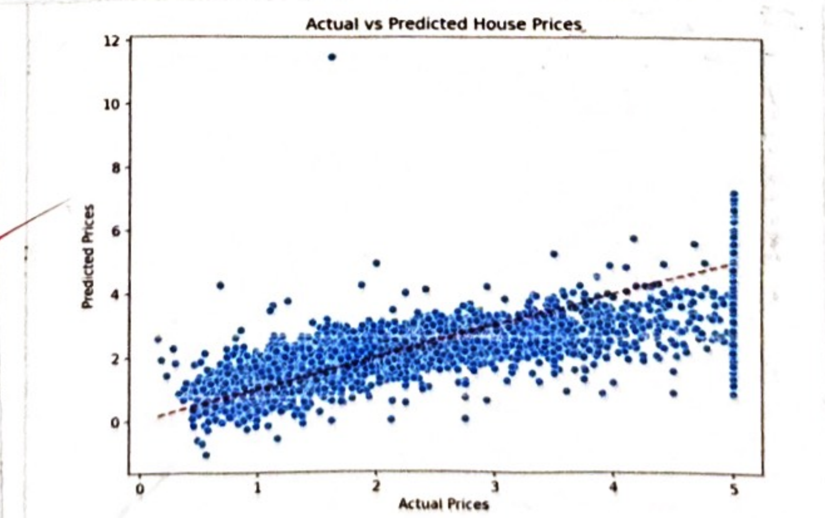
3. Linear regression : predict house prices using the bottom Housing dataset

```
import pandas as pd
import numpy as np
import Seaborn as sns
import matplotlib.pyplot as plt
Import sklearn.model_selection import train_test_split
 from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error , r2_score

#load california house dataset
 california = fetch_california_housing (as frame =True)
 df = california.frame
 print (df.head())

 #split data into features and target
 x= df.drop ('medHouseval , axis=1)
 Y= df ['medHouseval']
#split into training and test datasets
 X_train, x_test , Y_train , Y_test = train_test_split (x, y ,test_
size =0.2 , random_state =42)

 #create linear regression model
 model = LinearRegression ()
# train the model
```

```
model.fit (x_train , Y_train)
# predict on test set

Y_pred = model.predict (x_test)

# Evaluate the model

mse = mean_squared_error (y_test, Y_pred)

r2 = r2_score (y_test , y_pred)

print(f "mean Squared Error : {mse :.2f}")

print (f" R^2 Score : {r2:.2f}")

# ploat Actual vs predicted
plt.figure (figsize =(8,6))
sns.scatter plot (x=y_test ,y= y_pred)
plt.x label ("Actual prices")
~~plt.ylablel~~ plt.ylabel ("predicted price")
plt.titte ("Actual vs predicted House prices")
plt.plot ([min (y_test) , max (y_test), max (y_test)], [min (y_test),
  max (y_test)], '--r')
plt.show()
```


Actual vs Predicted House Prices
Predicted Prices
Actual Prices